

CLAIMS:

1. A method of generating an identification data block for a data carrier (41), which data carrier (41) comprises at least one track, wherein each track is defined by an item of start position information, wherein the identification data block is formed from part identification blocks by means of a gating function, wherein a first part identification block is formed from the items of start position information and a second part identification block is formed from a total for the number of tracks on the data carrier, characterized in that the first part identification block is formed from the items of start position information by means of an XOR gating operation and an XOR gating operation is then likewise used as a gating function.

10

2. A method as claimed in claim 1, wherein at least one track comprises a number of files having file names, use being made for generating the identification data block of, in addition, the file names to generate a third identification data block.

15

3. A method as claimed in claim 2, wherein characters of the file names are each individually gated by an XOR function.

20

4. A method as claimed in claim 2, wherein use is made for generating the identification data block of, in addition, a fourth part identification block, the total number of files, which is formed by the number of files, being used to generate the fourth identification data block.

5. A method as claimed in claim 1, wherein a data block having four bytes is generated as an identification data block.

25

6. A method as claimed in claim 5, wherein a data block having a single byte is generated as a second part identification data and, in the course of the XOR gating to generate the identification data block, this second part identification block is set to the fourth byte position in the identification data block.

7. A method as claimed in claim 5, wherein a data block having three bytes is generated as a first part identification block and, in the course of generating the identification data block, this second part identification block is set to the second byte position in the
5 identification data block.

8. An arrangement (10) for generating an identification data block for a data carrier (41), which data carrier (41) comprises at least one track, wherein each track is defined by an item of start position information, which arrangement (10) comprises the
10 means listed hereafter, namely determining means (51) for determining the item of start position information, gating means (59) for generating the identification data block by the gating of part identification blocks, first generating means (54) for generating a first part identification block from the items of start position information and second generating means (55) for generating a second part identification block from a total for the number of tracks on the data carrier, characterized in that the first generating means (54) are arranged to generate
15 the first part identification block by means of an XOR gating operation and in that the gating means (59) are arranged to generate the identification data block by means of an XOR function.

20 9. An arrangement (10) as claimed in claim 8, wherein third generating means (56) are provided that are arranged to generate a third part identification block from file names of files that are contained in the tracks on the data carrier.

10. An arrangement (10) as claimed in claim 9, wherein the third generating
25 means (56) are arranged to generate a third part identification block by means of an XOR gating operation.

11. An arrangement (10) as claimed in claim 8, wherein fourth generating means (57) are provided that are arranged to generate a fourth part identification block for
30 generating the identification data block, a total number of files that represents the number files that are contained in the tracks on the data carrier being used for this purpose.

12. A computer software product, which computer software product can be loaded directly into an internal storage means of a computer and comprises sections of software

code, in which case the method claimed in claim 1 can be carried out with the computer when the computer software product is run on the computer.

13. A computer software product as claimed in claim 12, wherein the product is
5 stored on a computer-readable medium.

14. A computer having a processing unit and an internal storage means, which
computer runs the computer software product claimed in claim 12.